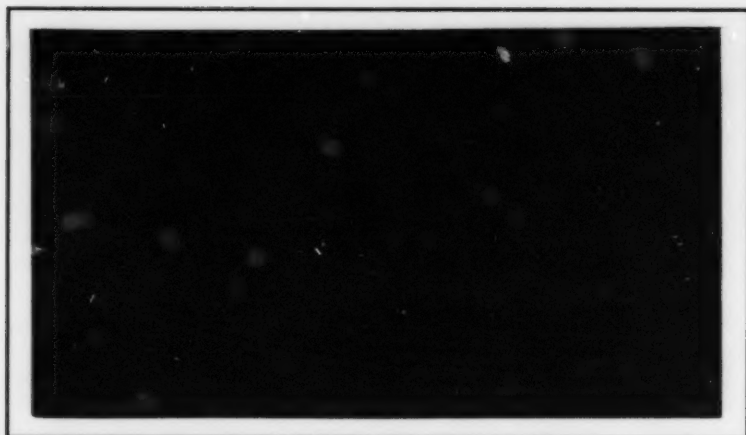




CENTRE FOR
SYSTEMS
SCIENCE

LCCR

**Local Tournaments and Proper
Circular Arc Graphs***

Pavol Hell, Jorgen Bang-Jensen, and Jing Huang

CSS/LCCR TR90-11

Revised August 1990

Local Tournaments and Proper Circular Arc Graphs*

Pavol HELL[†], Jørgen BANG-JENSEN[‡] and Jing HUANG[§]

Abstract

A *local tournament* is a digraph in which the out-set as well as the in-set of every vertex is a tournament. These digraphs have recently been found to share many desirable properties of tournaments. We illustrate this by giving $O(m + n \log n)$ algorithms to find a hamiltonian path and cycle in a local tournament. We mention several characterizations and recognition algorithms of graphs orientable as local tournaments. It turns out that they are precisely the graphs previously studied as proper circular arc graphs. Thus we obtain new recognition algorithms for proper circular arc graphs. We also give a more detailed structural characterization of chordal graphs that are orientable as local tournaments, i.e., that are proper circular arc graphs.

1 Local Tournaments

In [3], the second author introduced the class of locally semicomplete digraphs. For simplicity of this discussion we restrict the definition somewhat and define a *local tournament* to be an oriented digraph in which every vertex has any two out-neighbours adjacent and any two in-neighbours adjacent. In

*Partially supported by grants from NSERC and ASI. Most of this paper was written while the first two authors were visiting the Laboratoire de Recherche en Informatique, Université de Paris-Sud, whose hospitality is gratefully acknowledged.

[†]School of Computing Science, Simon Fraser University, Burnaby, B.C., Canada

[‡]Department of Computer Science, University of Copenhagen, Copenhagen, Denmark

[§]Dept. of Mathematics and Statistics, Simon Fraser University, Burnaby, B.C., Canada

other words, the out-set as well as the in-set of every vertex is a tournament. (A digraph is *oriented* if the adjacency relation is antisymmetric. A *locally semicomplete digraph* is defined just like a local tournament, except it is not necessarily oriented, i.e., for some u, v , both arcs uv and vu may be present. Our main purpose here is with orienting undirected graphs, and it is easy to see that a graph may be oriented to be locally semicomplete if and only if it may be oriented to be a local tournament.)

There are many examples of local tournaments; they include of course all tournaments, but also all directed paths and cycles, their powers, and other classes of digraphs. The main observation of [3] has been that many nice properties of tournaments extend to this larger class of digraphs. For instance, the strong components of a local tournament can be uniquely ordered $C_1, C_2, \dots, C_k$ so that each vertex of C_i dominates each vertex of C_{i+1} ($i = 1, \dots, k-1$) and no vertex of C_j dominates a vertex of C_i if $j > i$; moreover, if $k > 1$ then each C_i induces a tournament. On the other hand, if T is strong (i.e., if $k = 1$) then either T is a directed cycle, or there exists a vertex u such that $T - u$ is also strong. Many similar results are discussed in [3]. In particular it is proved there that every connected local tournament has a hamiltonian path and every strong tournament has a hamiltonian cycle. (Thus a local tournament has a hamiltonian path if and only if it is connected and has a hamiltonian cycle if and only if it is strong.)

Here we present two algorithms for constructing a hamiltonian (or, if not possible, then longest) path and cycle in a local tournament. They are both based on the following lemma, showing an efficient way to insert one new vertex into an already constructed path. We measure the complexity of the next two algorithms in the number of tests performed; a test consists of determining whether, for two given vertices u, v , it is the case that u dominates v , v dominates u , or there is no arc joining u and v . Although this measure ignores other costs of concern in actual implementation of the algorithm, one can achieve the claimed complexity by storing the growing path in a suitable data structure allowing logarithmic access and insertion, such as a balanced tree, cf. [11], 6.2.3.

Lemma 1 *Let T be a local tournament, let $P = x_1, x_2, \dots, x_r$ be a directed path of length r in T , and let x be a vertex of $T - P$ which is adjacent to a vertex x_i on P . With $O(\log r)$ tests we can find in T a directed path P' of length $r + 1$ containing the vertices $x, x_1, x_2, \dots, x_r$.*

Proof: Suppose that x dominates x_i (the proof in the other case is analogous). Clearly, if $i = 1$ then we may let $P' = x, x_1, \dots, x_r$. Hence we may assume that $i \geq 2$. Note that either xx_{i-1} or $x_{i-1}x$ is an arc of T , because T is a local tournament. It is easy to see by the same reasoning that either there exists a j , $1 \leq j \leq i-1$, such that x_jx and xx_{j+1} are both arcs of T , or xx_1 is an arc of T (this case may be viewed as having $j = 0$). In the latter case P' is defined above; in the former case we let P' be the path $x_1, x_2, \dots, x_j, x, x_{j+1}, \dots, x_r$. Thus it remains to find the j efficiently. This can be done by binary search. The first test we will perform concerns the vertices x and $x_{\lceil \frac{i}{2} \rceil}$. If there is no arc between $x_{\lceil \frac{i}{2} \rceil}$ and x , or if $x_{\lceil \frac{i}{2} \rceil}$ dominates x , then j is greater than or equal to $\lceil \frac{i}{2} \rceil$. Otherwise j is smaller than $\lceil \frac{i}{2} \rceil$. Thus continuing this way we find the j with at most $O(\log r)$ tests. $\square$

Theorem 1 *Suppose the vertices of the local tournament T are given in a fixed order $v_1, v_2, \dots, v_n$, together with a sequence $s(2), s(3), \dots, s(n)$ of subscripts such that each $s(i) < i$ and v_i has an arc to or from $v_{s(i)}$. Then a (directed) hamiltonian path in T can be found with $O(n \log n)$ tests.* $\square$

For practical implementation of this algorithm we must also consider the time required to obtain the ordering $v_1, \dots, v_n$ and the corresponding sequence $s(2), \dots, s(n)$. In general, it is not hard to see how to obtain both from a breadth first search of the underlying graph. Thus we have an algorithm of complexity $O(m + n \log n)$, provided the graph is represented in a form suitable to perform a test (defined above) in time $O(1)$ and to obtain all neighbours of v in time $\deg v$. (Here m is the number of arcs of T and $O(m)$ the time for breadth first search.) We may obtain such a representation from the standard adjacency lists by constructing, in time $O(m)$, a version of the adjacency matrix with valid entries certified by means of an additional stack, cf. [1], exercise 2.12.

If the local tournament is not connected, this will be revealed during the breadth first search and a longest path will be a hamiltonian path of a largest component. We summarize this as a corollary.

Corollary 1 *There is an $O(m + n \log n)$ algorithm to find a longest (directed) path in a local tournament.* $\square$

An alternate method to obtain the ordering $v_1, \dots, v_n$ and the sequence $s(2), \dots, s(n)$, would be to find a hamiltonian path of the underlying undirected graph. In corollary 5 we describe a situation when such a path can be

found in time $O(n \log n)$, hence yielding an $O(n \log n)$ algorithm for finding a directed hamiltonian path in a local tournament.

Theorem 2 *There is an $O(m + n \log n)$ algorithm for finding a (directed) hamiltonian cycle in a strong local tournament T .*

Proof: One can use the above hamiltonian path algorithm to do this. We offer the following algorithm, which is a generalization of an algorithm for tournaments by Manoussakis [12].

Using no more than $O(m + n \log n)$ steps we find a directed hamiltonian path $x_1, x_2, \dots, x_n$. Next we find x_i such that x_i dominates x_1 and i is as large as possible. If $i = n$ then $x_1, x_2, \dots, x_n, x_1$ is the desired cycle. Otherwise let $C = x_1, x_2, \dots, x_i, x_1$. Since $x_i x_{i+1}$ and $x_i x_1$ are arcs, either x_{i+1} dominates x_1 , or x_1 dominates x_{i+1} . In the former case x_{i+1} may be inserted in the cycle C (between x_i and x_1). In the latter case we consider the arcs $x_1 x_{i+1}$ and $x_1 x_2$. We find that either x_{i+1} can be inserted in C to give a directed cycle of length $i + 1$, or C completely dominates x_{i+1} . Suppose C completely dominates x_{i+1} , and $x_j, j > i$, dominates a vertex x_t of C . Then we obtain a longer cycle $x_1, \dots, x_{t-1}, x_{i+1}, x_{i+2}, \dots, x_j, x_t, x_{t+1}, \dots, x_i, x_1$. Continuing this way we always have a cycle $C' = x'_1, x'_2, \dots, x'_r, x'_1$ and a hamiltonian path $x'_1, x'_2, \dots, x'_n$. Since we have a strong digraph, it cannot happen that none of the vertices $x'_{r+1}, x'_{r+2}, \dots, x'_n$ dominates a vertex of C . Thus this process terminates with a hamiltonian cycle C' . Since we only consider each arc once in this part of the algorithm, we will have used no more than $O(m)$ new steps. $\square$

Corollary 2 *There exists an $O(m + n \log n)$ algorithm for finding a longest directed cycle in a local tournament.*

Proof: One can find all strong components, find a hamiltonian cycle in each, and choose the longest of them, all in time $O(m + n \log n)$. $\square$

Note that finding hamiltonian paths in transitive tournaments amounts to sorting and can be done with $O(n \log n)$ tests. $O(n \log n)$ algorithms for finding hamiltonian paths in *arbitrary* tournaments are given in [9], and the above algorithm further extends this to local tournaments. We also remark that (path-) merging in local tournaments can be achieved in linear time.

2 Proper Circular Arc Graphs

Here we study the class of undirected graphs that can be oriented as local tournaments. The main characterization is due to Skrien [14]. Formulated in our terminology, it says that the class in question consists precisely of proper circular arc graphs. A *circular arc graph* is the intersection graph of a family of arcs of a circle, i.e., a graph is a circular arc graph just if there exists a family of circular arcs indexed by the vertices of the graph such that two vertices are adjacent if and only if the two corresponding arcs intersect. A circular arc graph is *proper* if the family of arcs can be chosen to be inclusion-free, i.e., one is never entirely contained in another. Circular arc graphs have been found useful in a number of applications from cyclic scheduling to compiler design and psychometrics (cf. [16], [15]).

In order to describe the other characterizations we define, for an undirected graph $G = (V, E)$, the set E^* to consist of all *ordered* pairs uv where uv is an edge of G . The graph G^* has vertex set E^* and the ordered pair $uv \in E^*$ is adjacent to the ordered pair $vu \in E^*$, to any ordered pair $uw \in E^*$ with vw not an edge of G , to any ordered pair $wv \in E^*$ with uw not an edge of G , and to no other vertex of G^* .

Theorem 3 *The following statements are equivalent for a connected graph G :*

1. G is orientable as a local tournament
2. G is a proper circular arc graph
3. G does not contain as an induced subgraph the disjoint union of a cycle and an isolated vertex, or the tent (see figure 2) and an isolated vertex, or the complement of any of the graphs in figure 1
4. G^* is bipartite
5. G does not contain a sequence of edges $ab = u_1v_1, u_2v_2, \dots, u_kv_k = ba$ such that for each $i = 1, 2, \dots, k-1$, $u_i = u_{i+1}$ and v_i is not adjacent to v_{i+1} in G , or $v_i = v_{i+1}$ and u_i is not adjacent to u_{i+1} in G .

Proof: (Note that the edges $u_i v_i$ in 5. are best viewed as elements of E^* . It follows for instance that the claw, cf. figure 2, is excluded by 5.)

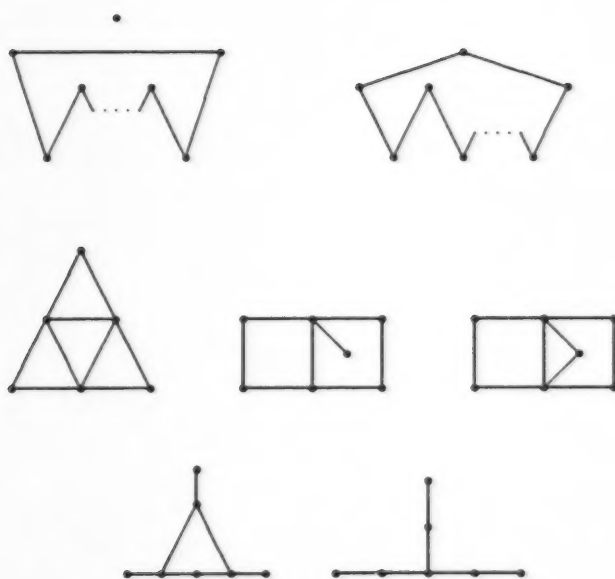


Figure 1: Subgraphs forbidden in the complement of G

The equivalence of the second and third statement is the result of Tucker [17]. One can derive the equivalence of the first two statements from this. In any event, this was first done by Skrien [14]. Specifically, the intersection graph of an inclusion-free family of circular arcs can be oriented by directing the edge uv from u to v if the leftmost point of the interval indexed by u precedes, in clockwise order, the leftmost point of the (intersecting) interval indexed by v . (It is possible to choose the arcs in such a way that no two share an endpoint or cover the entire circle, [6].) It is immediate that this orientation results in a local tournament. To prove the converse, that an underlying graph of a local tournament is a proper circular arc graph, it suffices to verify that none of the graphs in 3. of theorem 3 is orientable as a local tournament. This has been done by one of us [10], but we omit the details since we consequently discovered the reference [14]. To prove the equivalence of 1. and 4., assume that G is oriented as a local tournament T , and define a bipartition of E^* into E_1 , consisting of all pairs uv which are arcs of T , and E_2 , consisting of all pairs uv such that vu is an arc of T . It is easy to deduce, from the definitions, that both E_1 and E_2 are independent sets of G^* . Conversely, if E_1, E_2 is a bipartition of E^* into two independent sets of G^* , then E_1 contains precisely one of the pairs uv, vu for any edge uv of G . Orienting each edge according to its representative in E_1 results in a local tournament. The equivalence of 4. and 5. is best seen if one defines a binary relation Ψ on E^* by $uv\Psi vw$ if $uv, vw \in E^*$ but $uw \notin E$, $uv\Psi zu$ if $zu, uv \in E^*$ but $zv \notin E$, $uv\Psi uv$ if $uv \in E^*$, and no other pairs being related. It is clear that $uv\Psi u'v'$ just if uv is adjacent to $v'u'$ in G^* . Thus an odd cycle in G^* is the same object as the sequence $ab = u_1v_1, u_2v_2, \dots, u_kv_k = ba$ of 5. The transitive closure Ψ^* of Ψ is an equivalence relation, and 5. really says that no $ab\Psi^*ba$. $\square$

It is easy to see that G has precisely two orientations as a local tournament, one converse of the other, if and only if G^* is connected, and if and only if Ψ^* has exactly two equivalence classes.

Using the above theorem we have several choices for a concise certificate of a graph being a proper circular arc graph, or orientable as a local tournament: we can use an orientation as a local tournament, or a representation as the intersection graph of an inclusion-free family of circular arcs. In the same vein, for a concise certificate of not being a proper interval graph, or orientable as a local tournament, we may take the graphs of Figure 1, or an

odd cycle in G^* , or a sequence from 5. Finally, either 4. or 5. can serve as a method to recognize these graphs.

Corollary 3 *There is an $O(\Delta m)$ algorithm to decide if a given graph is orientable as a local tournament, and to find such an orientation if one exists.*

Proof: The graph G^* has $O(m)$ vertices and $O(\sum_{uv \in E} \deg u + \deg v) = O(\Delta m)$ edges and can be constructed in time $O(\Delta m)$. In the same time we can test whether it is bipartite by breadth first search. Any bipartition of G^* defines an orientation of G . This algorithm may also be implemented to work directly on G , by assigning an orientation to an edge uv , finding the equivalence class of uv under Ψ^* , etc. The complexity remains the same. $\square$

Corollary 4 *There is an $O(\Delta m)$ algorithm to decide if a given graph is a proper circular arc graph, and to find a representation by a proper family of circular arcs if one exists.* $\square$

Proof: We proceed as above, first constructing an orientation of G which is a local tournament with the additional property that the out-set and the in-set of every vertex is a *transitive* tournament. It is not hard to find a representation by a proper family of circular arcs for such a special local tournament. We have shown that to obtain such a special orientation of G it suffices to follow one simple rule when two-coloring G^* ; namely, whenever there is a choice (at the beginning, or after another component of G^* has been completely colored) we color white the uncolored vertex ij with the lexicographically smallest vector (i, j) . (Here we assume that $V(G) = 1, 2, \dots, n$.) The orientation of G^* arising from this two-coloring of G^* as in the proof of theorem 3, will be a local tournament with the additional property. (Note that this means that any local tournament may be re-oriented to be a local tournament with the additional property.) The details of this will appear elsewhere. $\square$

The classical algorithm for recognizing proper circular arc graphs, [18] has complexity $O(n^2)$. Our algorithm is more efficient for graphs with maximum degree $\Delta = o(\sqrt{n})$. Moreover, our algorithm doesn't depend on sophisticated data structures. On the other hand, using p, q -trees we have developed, with X. Deng, a linear algorithm for the same problem.

Corollary 5 *There is an $O(n \log n)$ algorithm to find a (directed) hamiltonian path in a connected local tournament for which we are given a circular arc representation of its underlying undirected graph.*

Proof: It follows from our earlier remarks that the underlying graph of a connected local tournament has an undirected hamiltonian path. Thus the algorithm of [5] will find one such path in time $O(n \log n)$ if the circular arc representation is given. The rest follows from the remarks following corollary 1. $\square$

As a last consequence of our theorem 3 we note that the following problems can also be efficiently solved on local tournaments:

1. Maximum independent set - in time $O(n \log n)$, if a circular arc representation of the underlying graph is given, and in time $O(n + \Delta m)$ otherwise [7]
2. Maximum clique - in time $O(n^2 \log \log n)$, [2]
3. Chromatic number - in time $O(n^{3/2} \log n)$, if a representation is given, [15]
4. Minimum cover by cliques - in time $O(n^2)$, [8]
5. Minimum dominating set and the domatic number - in time $O(nm)$ and $O(n^2 \log n)$ respectively, [4].

3 Chordal Graphs

In this section we obtain a more detailed structural characterization of graphs orientable as strong tournaments, i.e., proper circular arc graphs, for the case of chordal graphs. Our starting point is the following result of Roberts ([13]):

Theorem 4 *An interval graph G has an inclusion-free representation if and only if G is claw-free.* $\square$

This theorem in effect characterizes those proper circular arc graphs, and graphs orientable as local tournaments, that are interval graphs. The only forbidden subgraph is the claw, $K_{1,3}$. We have obtained a similar characterization for chordal graphs:



Figure 2: The claw, the net, and the tent

Theorem 5 *A chordal graph is orientable as a local tournament, i.e., is a proper circular arc graph, if and only if it is claw-free and net-free.*

A graph is *net-free* if it does not contain the net, the six-point graph obtained from K_3 by adding a pendant edge at each vertex.

Proof: It is immediate that neither the claw nor the net are orientable as local tournaments. Thus any G containing one of them is not orientable as a local tournament. On the other hand, suppose G is a net-free and claw-free chordal graph. Wegner, cf. [6], page 195, proved that this implies that G is an interval graph or it contains the tent – the sixpoint graph obtained from K_3 by joining each edge to a new vertex. If G is an interval graph, then theorem 4 assures that it is a proper circular arc graph. Suppose therefore that G contains the tent as an induced subgraph. We have proved, by a detailed analysis to appear elsewhere, that in this case G must be obtained from the tent by replacing each vertex with a complete graph (and making completely adjacent any two complete graphs corresponding to adjacent vertices of the tent). The tent is easily seen to be orientable as a local tournament (orient all arcs clockwise). On the other hand, any digraph obtained from a local tournament by substituting a tournament at each vertex is also a local tournament. Thus it suffices to orient all edges of G within the complete subgraphs arbitrarily, and all edges between the complete subgraphs according to the local tournament orientation of the tent; the result is an orientation of G as a local tournament. $\square$

References

- [1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, **The Design and Analysis of Computer Algorithms**, Addison-Wesley 1974.
- [2] A. Apostolico and S. E. Hambrusch, Finding maximum cliques on circular arc graphs, *Information Processing Letters* 26 (1987) 209 - 215.
- [3] J. Bang-Jensen, Locally semicomplete digraphs, *J. Graph Theory*, to appear.
- [4] M. A. Bonuccelli, Dominating sets and domatic number of circular arc graphs, *Discrete Applied Math.* 12 (1985) 203 - 213.
- [5] M. A. Bonuccelli and D. P. Bovet, Minimum disjoint path covering for circular arc graphs, *Information Processing Letters* 8 (1979) 159 - 161.
- [6] M. C. Golumbic, **Algorithmic Graph Theory and Perfect Graphs**, Academic Press 1980.
- [7] M. C. Golumbic and P. L. Hammer, Stability in circular arc graphs, *J. Algorithms* 9 (1988), 314 - 320.
- [8] U. I. Gupta, D. T. Lee, J. Y.-T. Leung, Efficient algorithms for interval graphs and circular arc graphs, *Networks* 12 (1982) 459 - 467.
- [9] P. Hell and M. Rosenfeld, The complexity of finding generalized paths in tournaments, *J. Algorithms* 4 (1983) 303 - 309.
- [10] J. Huang, A result on local tournaments, unpublished manuscript, 1989.
- [11] D. Knuth, **Art of Computer Programming**, v.3, Sorting and Searching, Addison-Wesley, 1973.
- [12] Y. Manoussakis, A linear algorithm for finding hamiltonian cycles in tournaments., Rapport de recherche 488, 1989, Laboratoire de Recherche en Informatique, Universite de Paris - Sud.
- [13] F. S. Roberts, Indifference graphs, in **Proof techniques in Graph Theory** (F. Harary ed.), pp. 139 - 146; Academic Press, 1969.

- [14] D. J. Skrien, A relationship between triangulated graphs, comparability graphs, proper interval graphs, proper circular arc graphs, and nested interval graphs, *J. Graph Theory* 6 (1982) 309-316.
- [15] A. Teng and A. Tucker, An $O(qn)$ algorithm to q-color a proper family of circular arcs, *Discrete Math.* 55 (1985) 233 - 243.
- [16] A. Tucker, Coloring a family of circular arcs, *SIAM J. Appl. Math.* 29 (1975) 493 - 502.
- [17] A. Tucker, Structure theorems for some circular arc graphs, *Discrete Math.* 7 (1974) 167 -195.
- [18] A. Tucker, Matrix characterizations of circular arc graphs, *Pacific J. Math.* 39 (1971) 535 - 545.

Centre for Systems Science

- Simon Fraser University established the **Centre for Systems Science** as a provincial centre for advanced study. The objective of CSS is twofold: coordinate and combine advanced systems research at SFU, and encourage application of research results within industry in order to further the technological development of British Columbia.
- CSS has assembled three research groups to streamline its efforts. The orientation of these groups is as follows: intelligent systems, microelectronics, and computer and communications systems. Because of shared research interests, this latter group merges with LCCR (**Laboratory for Computer and Communications Research**) for the publication of technical reports.
- CSS's three groups involve over 80 skilled faculty researchers principally from SFU's Schools of Engineering Science and Computing Science, but also draw substantially on Mathematics, Physics, Geography, Philosophy, Linguistics, Psychology, Communication, and Kinesiology.